

1. ALV VS. NON-ALV

**& Report ZRISHI_ALV_VS_NON_ALV*

report zrishi_alv_vs_non_alv no standard page heading.

data v_kunnr type kunnr.

selection-screen begin of block b1 with frame title text-001.

select-options s_kunnr for v_kunnr.

parameters: p_alv as checkbox,

p_nonalv as checkbox.

selection-screen end of block b1.

data: it_kna1 type table of kna1,

wa_kna1 type kna1.

select * into table it_kna1 from kna1 where kunnr in s_kunnr.

if p_nonalv = 'X'.

if it_kna1 is not initial.

write:/ 'Customer Num.' color 1,

15 'Country' color 2,

25 'Name' color 3,

45 'City' color 4.

uline.

loop at it_kna1 into wa_kna1.

write:/ wa_kna1-kunnr,

15 wa_kna1-land1,

25 wa_kna1-name1,

45 wa_kna1-ort01.

endloop.

endif.

endif.

if p_alv = 'X'.

call function 'REUSE_ALV_GRID_DISPLAY'

exporting

* I_INTERFACE_CHECK

= ' '

* I_BYPASSING_BUFFER

= ' '

* I_BUFFER_ACTIVE

= ' '

* I_CALLBACK_PROGRAM

= ' '

* I_CALLBACK_PF_STATUS_SET

= ' '

* I_CALLBACK_USER_COMMAND

= ' '

```

* I_CALLBACK_TOP_OF_PAGE           = ' '
* I_CALLBACK_HTML_TOP_OF_PAGE     = ' '
* I_CALLBACK_HTML_END_OF_LIST     = ' '
* i_structure_name                 = 'KNA1'
* I_BACKGROUND_ID                 = ' '
* I_GRID_TITLE                    =
* I_GRID_SETTINGS                 =
* IS_LAYOUT                       =
* IT_FIELDCAT                     =
* IT_EXCLUDING                    =
* IT_SPECIAL_GROUPS               =
* IT_SORT                         =
* IT_FILTER                       =
* IS_SEL_HIDE                     =
* I_DEFAULT                       = 'X'
* I_SAVE                          = ' '
* IS_VARIANT                      =
* IT_EVENTS                       =
* IT_EVENT_EXIT                   =
* IS_PRINT                        =
* IS_REPREP_ID                    =
* I_SCREEN_START_COLUMN           = 0
* I_SCREEN_START_LINE             = 0
* I_SCREEN_END_COLUMN             = 0
* I_SCREEN_END_LINE              = 0
* I_HTML_HEIGHT_TOP               = 0
* I_HTML_HEIGHT_END              = 0
* IT_ALV_GRAPHICS                 =
* IT_HYPERLINK                    =
* IT_ADD_FIELDCAT                 =
* IT_EXCEPT_QINFO               =
* IR_SALV_FULLSCREEN_ADAPTER      =
* IMPORTING
* E_EXIT_CAUSED_BY_CALLER         =
* ES_EXIT_CAUSED_BY_USER          =
tables
* t_outtab                        = it_kna1
* EXCEPTIONS
* PROGRAM_ERROR                   = 1
* OTHERS                          = 2

```

```
    if sy-subrc <> 0.  
    * Implement suitable error handling here  
    endif.
```

```
endif.
```

2. ALV USING MANUAL FIELD CATALOG

```
*& Report  ZRISHI_ALV_MANUAL
```

```
REPORT  zrishi_alv_manual.
```

```
TYPE-POOLS slis.
```

```
TYPES: BEGIN OF ty_kna1,  
        kunnr TYPE kunnr,  
        name1 TYPE name1,  
        land1 TYPE land1,  
        ort01 TYPE ort01,  
        prty TYPE zprty,  
        clr_row(4) TYPE c,  
        trf_light TYPE i,
```

```
        END OF ty_kna1.
```

```
DATA: it_kna1 TYPE TABLE OF ty_kna1,  
      it_fieldcat TYPE slis_t_fieldcat_alv,  
      wa_fieldcat TYPE slis_fieldcat_alv,  
      is_layout TYPE slis_layout_alv,  
      wa_kna1 TYPE ty_kna1.
```

```
DATA v_kunnr TYPE kunnr.
```

```
SELECT-OPTIONS s_kunnr FOR v_kunnr.
```

```
START-OF-SELECTION.
```

```
    SELECT kunnr name1 land1 ort01 prty INTO TABLE it_kna1 FROM  
    kna1 WHERE kunnr IN s_kunnr AND prty <> space.
```

```
END-OF-SELECTION.
```

```
    CLEAR wa_fieldcat.  
    wa_fieldcat-fieldname = 'KUNNR'.  
    wa_fieldcat-col_pos = 1.
```

```
wa_fieldcat-seltext_m = 'Customer No.'.
APPEND wa_fieldcat TO it_fieldcat.
```

```
CLEAR wa_fieldcat.
wa_fieldcat-fieldname = 'NAME1'.
wa_fieldcat-col_pos = 2.
wa_fieldcat-seltext_m = 'Name'.
APPEND wa_fieldcat TO it_fieldcat.
```

```
CLEAR wa_fieldcat.
wa_fieldcat-fieldname = 'LAND1'.
wa_fieldcat-col_pos = 3.
wa_fieldcat-seltext_m = 'Country'.
APPEND wa_fieldcat TO it_fieldcat.
```

```
CLEAR wa_fieldcat.
wa_fieldcat-fieldname = 'ORT01'.
wa_fieldcat-col_pos = 4.
wa_fieldcat-seltext_m = 'City'.
wa_fieldcat-hotspot = 'X'.
APPEND wa_fieldcat TO it_fieldcat.
```

```
CLEAR wa_fieldcat.
wa_fieldcat-fieldname = 'PTY'.
wa_fieldcat-col_pos = 5.
wa_fieldcat-seltext_m = 'Priority'.
APPEND wa_fieldcat TO it_fieldcat.
```

```
*CLEAR WA_FIELDCAT.
*      WA_FIELDCAT-HOTSPOT = 'X'.
*      MODIFY IT_FIELDCAT FROM WA_FIELDCAT TRANSPORTING EMPH
      ASIZE WHERE FIELDNAME = 'ORT01'.
```

```
is_layout-info_fieldname = 'CLR_ROW'.
CLEAR wa_kna1.
wa_kna1-clr_row = 'C71'.
MODIFY it_kna1 FROM wa_kna1 TRANSPORTING clr_row WHERE kunnr
= '0000001002'.
```

```
is_layout-lights_fieldname = 'TRF_LIGHT'.
```

```
CLEAR wa_kna1.
```

```
wa_kna1-trf_light = 1.
```

```
MODIFY it_kna1 FROM wa_kna1 TRANSPORTING trf_light  
WHERE prty = 'L'.
```

```
CLEAR wa_kna1.
```

```
wa_kna1-trf_light = 2.
```

```
MODIFY it_kna1 FROM wa_kna1 TRANSPORTING trf_light  
WHERE prty = 'M'.
```

```
CLEAR wa_kna1.
```

```
wa_kna1-trf_light = 3.
```

```
MODIFY it_kna1 FROM wa_kna1 TRANSPORTING trf_light  
WHERE prty = 'H'.
```

```
CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'  
EXPORTING
```

```
*      I_INTERFACE_CHECK           = ' '  
*      I_BYPASSING_BUFFER          = ' '  
*      I_BUFFER_ACTIVE             = ' '  
*      I_CALLBACK_PROGRAM          = ' '  
*      I_CALLBACK_PF_STATUS_SET    = ' '  
*      I_CALLBACK_USER_COMMAND     = ' '  
*      I_CALLBACK_TOP_OF_PAGE      = ' '  
*      I_CALLBACK_HTML_TOP_OF_PAGE = ' '  
*      I_CALLBACK_HTML_END_OF_LIST = ' '  
*      I_STRUCTURE_NAME            = 'KNA1'  
*      I_BACKGROUND_ID             = ' '  
*      I_GRID_TITLE                =  
*      I_GRID_SETTINGS             =  
*          is_layout                = is_layout  
*          it_fieldcat              = it_fieldcat  
*      IT_EXCLUDING                 =  
*      IT_SPECIAL_GROUPS            =  
*      IT_SORT                      =  
*      IT_FILTER                   =  
*      IS_SEL_HIDE                 =
```

```

*   I_DEFAULT                      = 'X'
*   I_SAVE                        = ' '
*   IS_VARIANT                    =
*   IT_EVENTS                     =
*   IT_EVENT_EXIT                 =
*   IS_PRINT                      =
*   IS_REPREP_ID                  =
*   I_SCREEN_START_COLUMN        = 0
*   I_SCREEN_START_LINE         = 0
*   I_SCREEN_END_COLUMN          = 0
*   I_SCREEN_END_LINE           = 0
*   I_HTML_HEIGHT_TOP            = 0
*   I_HTML_HEIGHT_END            = 0
*   IT_ALV_GRAPHICS               =
*   IT_HYPERLINK                  =
*   IT_ADD_FIELDCAT               =
*   IT_EXCEPT_QINFO             =
*   IR_SALV_FULLSCREEN_ADAPTER    =
* IMPORTING
*   E_EXIT_CAUSED_BY_CALLER       =
*   ES_EXIT_CAUSED_BY_USER       =
TABLES
    t_outtab                      = it_kna1
* EXCEPTIONS
*   PROGRAM_ERROR                 = 1
*   OTHERS                        = 2

```

```

IF sy-subrc <> 0.

```

```

* Implement suitable error handling here

```

```

ENDIF.

```

3. ALV USING AUTOMATIC FIELD CATALOG

```

*&-----
-----*

```

```

*& Report  ZRISHI_ALV_AUTO_FCATALOG

```

```

REPORT  zrishi_alv_auto_fcatalog.

```

```

DATA v_kunnr TYPE kunnr.

```

```

TYPE-POOLS slis.

```

```

SELECTION-SCREEN BEGIN OF BLOCK b1 WITH FRAME TITLE text-001.

```

```

SELECT-OPTIONS s_kunnr FOR v_kunnr.

```

```

SELECTION-SCREEN END OF BLOCK b1.

```

```

DATA: it_kna1 TYPE TABLE OF kna1,
      it_fieldcat TYPE slis_t_fieldcat_alv,
      wa_fieldcat TYPE LINE OF slis_t_fieldcat_alv.

```

```

SELECT * INTO TABLE it_kna1 FROM kna1 WHERE kunnr IN s_kunnr.

```

```

CALL FUNCTION 'REUSE_ALV_FIELDATALOG_MERGE'

```

```

  EXPORTING

```

```

*   I_PROGRAM_NAME                =
*   I_INTERNAL_TABNAME            =
*   i_structure_name              = 'KNA1'
*   I_CLIENT_NEVER_DISPLAY        = 'X'
*   I_INCLNAME                    =
*   I_BYPASSING_BUFFER            =
*   I_BUFFER_ACTIVE               =

```

```

  CHANGING

```

```

    ct_fieldcat                  = it_fieldcat
* EXCEPTIONS
*   INCONSISTENT_INTERFACE        = 1
*   PROGRAM_ERROR                 = 2
*   OTHERS                        = 3

```

```

IF sy-subrc <> 0.

```

```

  * Implement suitable error handling here

```

```

ENDIF.

```

```

CLEAR wa_fieldcat.

```

```

wa_fieldcat-emphasize = 'C61'.

```

```

MODIFY it_fieldcat FROM wa_fieldcat TRANSPORTING emphasize WHE
RE fieldname = 'ORT01'.

```

```

CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'

```

```

  EXPORTING

```

```

*   I_INTERFACE_CHECK            = ' '
*   I_BYPASSING_BUFFER            = ' '
*   I_BUFFER_ACTIVE               = ' '
*   I_CALLBACK_PROGRAM            = ' '
*   I_CALLBACK_PF_STATUS_SET      = ' '

```

```

* I_CALLBACK_USER_COMMAND           = ' '
* I_CALLBACK_TOP_OF_PAGE           = ' '
* I_CALLBACK_HTML_TOP_OF_PAGE      = ' '
* I_CALLBACK_HTML_END_OF_LIST      = ' '
i_structure_name                    = 'KNA1'
* I_BACKGROUND_ID                  = ' '
* I_GRID_TITLE                     =
* I_GRID_SETTINGS                  =
* IS_LAYOUT                        =
it_fieldcat                         = it_fieldcat
* IT_EXCLUDING                     =
* IT_SPECIAL_GROUPS                =
* IT_SORT                          =
* IT_FILTER                        =
* IS_SEL_HIDE                      =
* I_DEFAULT                        = 'X'
* I_SAVE                           = ' '
* IS_VARIANT                       =
* IT_EVENTS                        =
* IT_EVENT_EXIT                    =
* IS_PRINT                         =
* IS_REPREP_ID                     =
* I_SCREEN_START_COLUMN            = 0
* I_SCREEN_START_LINE              = 0
* I_SCREEN_END_COLUMN              = 0
* I_SCREEN_END_LINE                = 0
* I_HTML_HEIGHT_TOP                 = 0
* I_HTML_HEIGHT_END                 = 0
* IT_ALV_GRAPHICS                  =
* IT_HYPERLINK                     =
* IT_ADD_FIELDCAT                  =
* IT_EXCEPT_QINFO                 =
* IR_SALV_FULLSCREEN_ADAPTER        =
* IMPORTING
* E_EXIT_CAUSED_BY_CALLER           =
* ES_EXIT_CAUSED_BY_USER            =
TABLES
  t_outtab                         = it_kna1
* EXCEPTIONS
* PROGRAM_ERROR                     = 1

```



```

        15 'Country' COLOR 2,
        25 'Name' COLOR 3,
        45 'City' COLOR 4.

    ULINE.
    LOOP AT it_kna1 INTO wa_kna1.
        WRITE:/ wa_kna1-kunnr,
            15 wa_kna1-land1,
            25 wa_kna1-name1,
            45 wa_kna1-ort01.

    ENDLOOP.
ENDIF.
ENDIF.

IF p_alv = 'X'.
    IF p_auto = 'X'.
        CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
            EXPORTING
*         I_INTERFACE_CHECK                = ' '
*         I_BYPASSING_BUFFER              = ' '
*         I_BUFFER_ACTIVE                 = ' '
*         I_CALLBACK_PROGRAM              = ' '
*         I_CALLBACK_PF_STATUS_SET        = ' '
*         I_CALLBACK_USER_COMMAND         = ' '
*         I_CALLBACK_TOP_OF_PAGE          = ' '
*         I_CALLBACK_HTML_TOP_OF_PAGE     = ' '
*         I_CALLBACK_HTML_END_OF_LIST     = ' '
            i_structure_name                = 'KNA1'
*         I_BACKGROUND_ID                 = ' '
*         I_GRID_TITLE                    =
*         I_GRID_SETTINGS                 =
*         IS_LAYOUT                       =
*         IT_FIELDCAT                     =
*         IT_EXCLUDING                    =
*         IT_SPECIAL_GROUPS               =
*         IT_SORT                         =
*         IT_FILTER                       =
*         IS_SEL_HIDE                     =
*         I_DEFAULT                       = 'X'
*         I_SAVE                          = ' '
*         IS_VARIANT                      =

```

```

*      IT_EVENTS                      =
*      IT_EVENT_EXIT                  =
*      IS_PRINT                        =
*      IS_REPREP_ID                   =
*      I_SCREEN_START_COLUMN          = 0
*      I_SCREEN_START_LINE            = 0
*      I_SCREEN_END_COLUMN            = 0
*      I_SCREEN_END_LINE              = 0
*      I_HTML_HEIGHT_TOP              = 0
*      I_HTML_HEIGHT_END              = 0
*      IT_ALV_GRAPHICS                =
*      IT_HYPERLINK                   =
*      IT_ADD_FIELDCAT                =
*      IT_EXCEPT_QINFO               =
*      IR_SALV_FULLSCREEN_ADAPTER      =
* IMPORTING
*      E_EXIT_CAUSED_BY_CALLER         =
*      ES_EXIT_CAUSED_BY_USER          =
      TABLES
          t_outtab                      = it_kna1
* EXCEPTIONS
*      PROGRAM_ERROR                   = 1
*      OTHERS                          = 2
      .
ENDIF.
IF sy-subrc <> 0.
* Implement suitable error handling here
ENDIF.
IF p_semi = 'X'.
    CALL FUNCTION 'REUSE_ALV_FIELDCATALOG_MERGE'
    EXPORTING
*      I_PROGRAM_NAME                  =
*      I_INTERNAL_TABNAME              =
*      i_structure_name                 = 'KNA1'
*      I_CLIENT_NEVER_DISPLAY           = 'X'
*      I_INCLNAME                      =
*      I_BYPASSING_BUFFER               =
*      I_BUFFER_ACTIVE                 =
    CHANGING
        ct_fieldcat                    = it_fieldcat

```

```

*      EXCEPTIONS
*      INCONSISTENT_INTERFACE          = 1
*      PROGRAM_ERROR                   = 2
*      OTHERS                          = 3
*
*      IF sy-subrc <> 0.
*      Implement suitable error handling here
*      ENDIF.
*      CLEAR wa_fieldcat.
*      wa_fieldcat-emphasize = 'C71'.
*      MODIFY it_fieldcat FROM wa_fieldcat TRANSPORTING emphasize
*                                          WHERE fieldname = 'NAM
E1'.
*      CLEAR wa_fieldcat.
*      wa_fieldcat-emphasize = 'C61'.
*      MODIFY it_fieldcat FROM wa_fieldcat TRANSPORTING emphasize
*                                          WHERE fieldname = 'ORT
01'.

CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
EXPORTING
*      I_INTERFACE_CHECK              = ' '
*      I_BYPASSING_BUFFER              = ' '
*      I_BUFFER_ACTIVE                 = ' '
*      I_CALLBACK_PROGRAM              = ' '
*      I_CALLBACK_PF_STATUS_SET        = ' '
*      I_CALLBACK_USER_COMMAND         = ' '
*      I_CALLBACK_TOP_OF_PAGE          = ' '
*      I_CALLBACK_HTML_TOP_OF_PAGE     = ' '
*      I_CALLBACK_HTML_END_OF_LIST     = ' '
*      I_STRUCTURE_NAME                = 'KNA1'
*      I_BACKGROUND_ID                 = ' '
*      I_GRID_TITLE                    =
*      I_GRID_SETTINGS                 =
*      IS_LAYOUT                       =
*      it_fieldcat                     = it_fieldcat
*      IT_EXCLUDING                    =
*      IT_SPECIAL_GROUPS               =
*      IT_SORT                         =
*      IT_FILTER                       =

```

```

*      IS_SEL_HIDE                        =
*      I_DEFAULT                         = 'X'
*      I_SAVE                           = ' '
*      IS_VARIANT                        =
*      IT_EVENTS                        =
*      IT_EVENT_EXIT                    =
*      IS_PRINT                         =
*      IS_REPREP_ID                     =
*      I_SCREEN_START_COLUMN            = 0
*      I_SCREEN_START_LINE              = 0
*      I_SCREEN_END_COLUMN              = 0
*      I_SCREEN_END_LINE                = 0
*      I_HTML_HEIGHT_TOP                = 0
*      I_HTML_HEIGHT_END                = 0
*      IT_ALV_GRAPHICS                  =
*      IT_HYPERLINK                     =
*      IT_ADD_FIELDCAT                  =
*      IT_EXCEPT_QINFO                 =
*      IR_SALV_FULLSCREEN_ADAPTER       =
*  IMPORTING
*      E_EXIT_CAUSED_BY_CALLER          =
*      ES_EXIT_CAUSED_BY_USER           =
TABLES
      t_outtab                          = it_kna1
*  EXCEPTIONS
*      PROGRAM_ERROR                    = 1
*      OTHERS                           = 2
.
  IF sy-subrc <> 0.
*  Implement suitable error handling here
  ENDIF.
ENDIF.
ENDIF.

```

5. ALV CELL COLORING PROGRAM

```

*&-----
-----*
*& Report  ZRISHI_ALV_MANUAL
REPORT  zrishi_alv_cell_coloring.

```

```
TYPE-POOLS slis.
```

```
DATA v_kunnr TYPE kunnr.
```

```
SELECT-OPTIONS s_kunnr FOR v_kunnr.
```

```
TYPES: BEGIN OF ty_kna1,  
        kunnr TYPE kunnr,  
        name1 TYPE name1,  
        land1 TYPE land1,  
        ort01 TYPE ort01,  
        prty TYPE zprty,  
        clr_row(4) TYPE c,  
        trf_light TYPE i,  
        cell_tab TYPE slis_t_specialcol_alv ,  
    END OF ty_kna1.
```

```
TYPES: BEGIN OF ty_temp_kna1,  
        kunnr TYPE kunnr,  
        name1 TYPE name1,  
        land1 TYPE land1,  
        ort01 TYPE ort01,  
        prty TYPE zprty,  
    END OF ty_temp_kna1.
```

```
DATA: it_kna1 TYPE TABLE OF ty_kna1,  
      it_fieldcat TYPE slis_t_fieldcat_alv,  
      it_temp_kna1 TYPE TABLE OF ty_temp_kna1,  
      itab TYPE slis_t_specialcol_alv ,  
      wa_kna1 TYPE ty_kna1,  
      wa_fieldcat TYPE slis_fieldcat_alv,  
      wa_temp_kna1 TYPE ty_temp_kna1,  
      wa TYPE LINE OF slis_t_specialcol_alv ,  
      is_layout TYPE slis_layout_alv.
```

```
START-OF-SELECTION.
```

```
SELECT kunnr name1 land1 ort01 prty INTO TABLE it_temp_kna1 FR  
OM kna1 WHERE kunnr IN s_kunnr AND prty <> space.
```

**Move Data from IT_TEMP_KNA1 To IT_KNA1*

```
LOOP AT it_temp_kna1 INTO wa_temp_kna1.  
  wa_kna1-kunnr = wa_temp_kna1-kunnr .  
  wa_kna1-name1 = wa_temp_kna1-name1 .  
  wa_kna1-land1 = wa_temp_kna1-land1 .  
  wa_kna1-ort01 = wa_temp_kna1-ort01 .  
  wa_kna1-prty = wa_temp_kna1-prty .  
  APPEND wa_kna1 TO it_kna1 .  
ENDLOOP.
```

END-OF-SELECTION.

```
CLEAR wa_fieldcat.  
wa_fieldcat-fieldname = 'KUNNR'.  
wa_fieldcat-col_pos = 1.  
wa_fieldcat-seltext_m = 'Customer No.'.  
APPEND wa_fieldcat TO it_fieldcat.
```

```
CLEAR wa_fieldcat.  
wa_fieldcat-fieldname = 'NAME1'.  
wa_fieldcat-col_pos = 2.  
wa_fieldcat-seltext_m = 'Name'.  
APPEND wa_fieldcat TO it_fieldcat.
```

```
CLEAR wa_fieldcat.  
wa_fieldcat-fieldname = 'LAND1'.  
wa_fieldcat-col_pos = 3.  
wa_fieldcat-seltext_m = 'Country'.  
APPEND wa_fieldcat TO it_fieldcat.
```

```
CLEAR wa_fieldcat.  
wa_fieldcat-fieldname = 'ORT01'.  
wa_fieldcat-col_pos = 4.  
wa_fieldcat-seltext_m = 'City'.  
wa_fieldcat-hotspot = 'X'.  
APPEND wa_fieldcat TO it_fieldcat.
```

```
CLEAR wa_fieldcat.  
wa_fieldcat-fieldname = 'PRTY'.  
wa_fieldcat-col_pos = 5.
```

```
wa_fieldcat-seltext_m = 'Priority'.  
APPEND wa_fieldcat TO it_fieldcat.
```

```
*CLEAR WA_FIELDCAT.  
*      WA_FIELDCAT-HOTSPOT = 'X'.  
*      MODIFY IT_FIELDCAT FROM WA_FIELDCAT TRANSPORTING EMPH  
ASIZE WHERE FIELDNAME = 'ORT01'.
```

```
is_layout-info_fieldname = 'CLR_ROW'.  
CLEAR wa_kna1.  
wa_kna1-clr_row = 'C71'.  
MODIFY it_kna1 FROM wa_kna1 TRANSPORTING clr_row WHERE kunnr  
= '0000001002'.
```

```
is_layout-lights_fieldname = 'TRF_LIGHT'.
```

```
CLEAR wa_kna1.  
wa_kna1-trf_light = 1.  
MODIFY it_kna1 FROM wa_kna1 TRANSPORTING trf_light  
WHERE prty = 'L'.
```

```
CLEAR wa_kna1.  
wa_kna1-trf_light = 2.  
MODIFY it_kna1 FROM wa_kna1 TRANSPORTING trf_light  
WHERE prty = 'M'.
```

```
CLEAR wa_kna1.  
wa_kna1-trf_light = 3.  
MODIFY it_kna1 FROM wa_kna1 TRANSPORTING trf_light  
WHERE prty = 'H'.
```

```
is_layout-coltab_fieldname = 'CELL_TAB'.
```

```
CLEAR wa.  
wa-fieldname = 'PTY'.  
wa-color-col = 6.  
wa-color-int = 1.  
APPEND wa TO itab.
```



```

*   IS_PRINT                        =
*   IS_REPREP_ID                   =
*   I_SCREEN_START_COLUMN          = 0
*   I_SCREEN_START_LINE            = 0
*   I_SCREEN_END_COLUMN            = 0
*   I_SCREEN_END_LINE              = 0
*   I_HTML_HEIGHT_TOP              = 0
*   I_HTML_HEIGHT_END              = 0
*   IT_ALV_GRAPHICS                =
*   IT_HYPERLINK                   =
*   IT_ADD_FIELDCAT                =
*   IT_EXCEPT_QINFO              =
*   IR_SALV_FULLSCREEN_ADAPTER     =
* IMPORTING
*   E_EXIT_CAUSED_BY_CALLER        =
*   ES_EXIT_CAUSED_BY_USER        =
TABLES
    t_outtab                       = it_kna1
* EXCEPTIONS
*   PROGRAM_ERROR                  = 1
*   OTHERS                         = 2
.
IF sy-subrc <> 0.
* Implement suitable error handling here
ENDIF.

```

6.ALV EVENT HANDLING USING FM

```

*&-----
-----*
*& Report  ZRISHI_ALV_EVENT_HANDLING
*&
*&-----
-----*
*&
*&
*&-----
-----*

```

REPORT zrishi_alv_event_handling.

```
TYPE-POOLS slis.
```

```
DATA v_kunnr TYPE kunnr.
```

```
SELECT-OPTIONS s_kunnr FOR v_kunnr.
```

```
TYPES: BEGIN OF ty_kna1,  
        kunnr TYPE kunnr,  
        name1 TYPE name1,  
        land1 TYPE land1,  
        ort01 TYPE ort01,  
        row_clr(4) TYPE c,  
        END OF ty_kna1.
```

```
DATA: it_kna1 TYPE TABLE OF ty_kna1,  
      wa_kna1 TYPE ty_kna1,  
      it_fieldcat TYPE slis_t_fieldcat_alv,  
      wa_fieldcat TYPE LINE OF slis_t_fieldcat_alv,  
      it_events TYPE slis_t_event,  
      wa_events TYPE LINE OF slis_t_event,  
      is_layout TYPE slis_layout_alv.
```

```
START-OF-SELECTION.
```

```
    SELECT kunnr name1 land1 ort01 INTO TABLE it_kna1 FROM kna1  
    WHERE kunnr IN s_kunnr.
```

```
    CLEAR wa_events.  
    wa_events-name = 'TOP_OF_PAGE'.  
    wa_events-form = 'GET_CUST_DATA'.  
    APPEND wa_events TO it_events.
```

```
END-OF-SELECTION.
```

```
    CLEAR wa_fieldcat.  
    wa_fieldcat-fieldname = 'KUNNR'.  
    wa_fieldcat-col_pos = '1'.  
    wa_fieldcat-seltext_m = 'Customer No.'.  
    APPEND wa_fieldcat TO it_fieldcat.
```

```
    CLEAR wa_fieldcat.  
    wa_fieldcat-fieldname = 'NAME1'.  
    wa_fieldcat-col_pos = 2.
```

```
wa_fieldcat-seltext_m = 'Name'.
APPEND wa_fieldcat TO it_fieldcat.
```

```
CLEAR wa_fieldcat.
wa_fieldcat-fieldname = 'LAND1'.
wa_fieldcat-col_pos = 3.
wa_fieldcat-seltext_m = 'Country'.
APPEND wa_fieldcat TO it_fieldcat.
```

```
CLEAR wa_fieldcat.
wa_fieldcat-fieldname = 'ORT01'.
wa_fieldcat-col_pos = 4.
wa_fieldcat-seltext_m = 'City'.
wa_fieldcat-hotspot = 'X'.
APPEND wa_fieldcat TO it_fieldcat.
```

```
is_layout-info_fieldname = 'ROW_CLR'.
```

```
CLEAR wa_kna1.
wa_kna1-row_clr = 'C71'.
MODIFY it_kna1 FROM wa_kna1 TRANSPORTING row_clr WHERE kunnr
= '0000001000'.
```

```
CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
EXPORTING
```

```
*      I_INTERFACE_CHECK                = ' '
*      I_BYPASSING_BUFFER                = ' '
*      I_BUFFER_ACTIVE                   = ' '
*      i_callback_program                 = sy-cprog
*      I_CALLBACK_PF_STATUS_SET           = ' '
*      I_CALLBACK_USER_COMMAND            = ' '
*      I_CALLBACK_TOP_OF_PAGE             = ' '
*      I_CALLBACK_HTML_TOP_OF_PAGE        = ' '
*      I_CALLBACK_HTML_END_OF_LIST        = ' '
*      I_STRUCTURE_NAME                   =
*      I_BACKGROUND_ID                    = ' '
*      I_GRID_TITLE                       =
*      I_GRID_SETTINGS                    =
*      is_layout                          = is_layout
*      it_fieldcat                        = it_fieldcat
```

```

*   IT_EXCLUDING                      =
*   IT_SPECIAL_GROUPS                =
*   IT_SORT                          =
*   IT_FILTER                        =
*   IS_SEL_HIDE                      =
*   I_DEFAULT                        = 'X'
*   I_SAVE                          = ' '
*   IS_VARIANT                      =
it_events                          = it_events
*   IT_EVENT_EXIT                    =
*   IS_PRINT                        =
*   IS_REPREP_ID                    =
*   I_SCREEN_START_COLUMN            = 0
*   I_SCREEN_START_LINE              = 0
*   I_SCREEN_END_COLUMN              = 0
*   I_SCREEN_END_LINE                = 0
*   I_HTML_HEIGHT_TOP                = 0
*   I_HTML_HEIGHT_END                = 0
*   IT_ALV_GRAPHICS                  =
*   IT_HYPERLINK                     =
*   IT_ADD_FIELDCAT                  =
*   IT_EXCEPT_QINFO                =
*   IR_SALV_FULLSCREEN_ADAPTER        =
* IMPORTING
*   E_EXIT_CAUSED_BY_CALLER          =
*   ES_EXIT_CAUSED_BY_USER           =
TABLES
    t_outtab                        = it_knal
* EXCEPTIONS
*   PROGRAM_ERROR                    = 1
*   OTHERS                           = 2

```

IF sy-subrc <> 0.

```

* Implement suitable error handling here
ENDIF.

```

```

*&-----
-----*
*&      Form  GET_CUST_DATA
*&-----

```

```

-----*
*           text
*-----
-----*

FORM get_cust_data.
  DATA: it_list TYPE slis_t_listheader,
         wa_list TYPE LINE OF slis_t_listheader.

  CLEAR wa_list.
  wa_list-typ = 'H'.
  wa_list-info = 'GET THE DETAILS OF CUSTOMER'.
  APPEND wa_list TO it_list.

  CLEAR wa_list.
  wa_list-typ = 'S'.
  wa_list-info = 'GET THE DETAILS OF CUSTOMER'.
  APPEND wa_list TO it_list.

  CLEAR wa_list.
  wa_list-typ = 'A'.
  wa_list-info = 'GET THE DETAILS OF CUSTOMER'.
  APPEND wa_list TO it_list.

  CALL FUNCTION 'REUSE_ALV_COMMENTARY_WRITE'
    EXPORTING
      it_list_commentary = it_list
      i_logo              = 'ENJOYSAP_LOGO'
*      I_END_OF_LIST_GRID =
*      I_ALV_FORM         =
      .

ENDFORM.                    "GET_CUST_DATA

7.00 ALV PROGRAM
*&-----
-----*
*& Report  ZRISHI_00_ALV
REPORT  zrishi_oo_alv.
DATA : o_container TYPE REF TO cl_gui_custom_container,
      o_grid       TYPE REF TO cl_gui_alv_grid,

```

```

        it_kna1 TYPE TABLE OF kna1.
*-----*
*          CLASS lcl_event_receiver DEFINITION
*-----*
*
*-----*
*-----*
CLASS lcl_event_receiver DEFINITION.
  PUBLIC SECTION.
    METHODS : handle_double_click FOR EVENT double_click
              OF cl_gui_alv_grid IMPORTING e_row e_column,

              handle_toolbar FOR EVENT toolbar
              OF cl_gui_alv_grid IMPORTING e_object,

              handle_user_command FOR EVENT user_command
              OF cl_gui_alv_grid IMPORTING e_ucomm.

ENDCLASS.                                "lcl_event_receiver DEFINITION
*Implement Event Handler Class
CLASS lcl_event_receiver IMPLEMENTATION.

  METHOD handle_user_command.
    DATA it_lfa1 TYPE TABLE OF lfa1.
    CASE e_ucomm.
      WHEN 'GETAI'.
        IF it_kna1 IS NOT INITIAL .
          SELECT * INTO TABLE it_lfa1 FROM lfa1 FOR ALL ENTRI
ES IN it_kna1
                                WHERE kunnr = it_kn
a1-kunnr.

          CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
            EXPORTING

*          I_INTERFACE_CHECK                = ' '
*          I_BYPASSING_BUFFER              = ' '
*          I_BUFFER_ACTIVE                 = ' '
*          I_CALLBACK_PROGRAM              = ' '
*          I_CALLBACK_PF_STATUS_SET        = ' '

```

```

*          I_CALLBACK_USER_COMMAND          = ' '
*          I_CALLBACK_TOP_OF_PAGE           = ' '
*          I_CALLBACK_HTML_TOP_OF_PAGE      = ' '
*          I_CALLBACK_HTML_END_OF_LIST      = ' '
i_structure_name = 'LFA1'
*          I_BACKGROUND_ID                  = ' '
*          I_GRID_TITLE                     =
*          I_GRID_SETTINGS                  =
*          IS_LAYOUT                         =
*          IT_FIELDCAT                      =
*          IT_EXCLUDING                     =
*          IT_SPECIAL_GROUPS                =
*          IT_SORT                          =
*          IT_FILTER                        =
*          IS_SEL_HIDE                      =
*          I_DEFAULT                        = 'X'
*          I_SAVE                          = '
,
*          IS_VARIANT                       =
*          IT_EVENTS                        = IT_
EVENTS
*          IT_EVENT_EXIT                    =
*          IS_PRINT                         =
*          IS_REPREP_ID                     =
*          I_SCREEN_START_COLUMN            = 0
*          I_SCREEN_START_LINE              = 0
*          I_SCREEN_END_COLUMN              = 0
*          I_SCREEN_END_LINE                = 0
*          I_HTML_HEIGHT_TOP                = 0
*          I_HTML_HEIGHT_END                = 0
*          IT_ALV_GRAPHICS                   =
*          IT_HYPERLINK                     =
*          IT_ADD_FIELDCAT                  =
*          IT_EXCEPT_QINFO                 =
*          IR_SALV_FULLSCREEN_ADAPTER        =
*          IMPORTING
*          E_EXIT_CAUSED_BY_CALLER          =
*          ES_EXIT_CAUSED_BY_USER           =

```

TABLES

```

t_outtab = it_lfal

```



```

*              EXCEPTIONS
*              PROGRAM_ERROR                = 1
*              OTHERS                      = 2
*
*              .
*              IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*              WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
*              ENDIF.
*              ENDIF.
*              ENDCASE.
*              ENDMETHOD.                  "handle_user_command

METHOD handle_toolbar.
  DATA wa_button TYPE LINE OF ttb_button.
  wa_button-function = 'GETAI'.
  wa_button-text = 'GetAllItems'.
  APPEND wa_button TO e_object->mt_toolbar.

ENDMETHOD.                  "handle_toolbar

METHOD handle_double_click.
  DATA : wa_kna1 TYPE kna1,
          it_lfa1 TYPE TABLE OF lfa1.
  READ TABLE it_kna1 INTO wa_kna1 INDEX e_row-index.

  IF e_column = 'KUNNR'.
    SELECT * INTO TABLE it_lfa1 FROM lfa1
          WHERE kunnr = wa_kna1-kunnr.

    CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
      EXPORTING
*              I_INTERFACE_CHECK                = ' '
*              I_BYPASSING_BUFFER              = ' '
*              I_BUFFER_ACTIVE                 = ' '
*              I_CALLBACK_PROGRAM              = ' '
*              I_CALLBACK_PF_STATUS_SET        = ' '
*              I_CALLBACK_USER_COMMAND         = ' '
*              I_CALLBACK_TOP_OF_PAGE          = ' '
*              I_CALLBACK_HTML_TOP_OF_PAGE     = ' '
*              I_CALLBACK_HTML_END_OF_LIST     = ' '

```

```

i_structure_name = 'LFA1'

*      I_BACKGROUND_ID = ' '
*      I_GRID_TITLE =
*      I_GRID_SETTINGS =
*      IS_LAYOUT =
*      IT_FIELDCAT =
*      IT_EXCLUDING =
*      IT_SPECIAL_GROUPS =
*      IT_SORT =
*      IT_FILTER =
*      IS_SEL_HIDE =
*      I_DEFAULT = 'X'
*      I_SAVE = '
,
*      IS_VARIANT =
*      IT_EVENTS = IT_
EVENTS
*      IT_EVENT_EXIT =
*      IS_PRINT =
*      IS_REPREP_ID =
*      I_SCREEN_START_COLUMN = 0
*      I_SCREEN_START_LINE = 0
*      I_SCREEN_END_COLUMN = 0
*      I_SCREEN_END_LINE = 0
*      I_HTML_HEIGHT_TOP = 0
*      I_HTML_HEIGHT_END = 0
*      IT_ALV_GRAPHICS =
*      IT_HYPERLINK =
*      IT_ADD_FIELDCAT =
*      IT_EXCEPT_QINFO =
*      IR_SALV_FULLSCREEN_ADAPTER =
*      IMPORTING
*      E_EXIT_CAUSED_BY_CALLER =
*      ES_EXIT_CAUSED_BY_USER =

```

TABLES

```

t_outtab = it_lfa1
*      EXCEPTIONS
*      PROGRAM_ERROR = 1
*      OTHERS = 2
.

```

```

        IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*       WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
        ENDIF.
    ENDIF.

    ENDMETHOD.                "handle_double_click
ENDCLASS.                    "lcl_event_receiver IMPLEMENTATI
ON
DATA o_event_rcvr TYPE REF TO lcl_event_receiver.

START-OF-SELECTION.
    CALL SCREEN 9000.

*-----
*-----*
*  MODULE READ_DATA OUTPUT
*-----
*-----*
*
*-----
*-----*

MODULE read_data OUTPUT.
    SELECT * INTO TABLE it_kna1 FROM kna1 WHERE land1 = 'US'.

ENDMODULE.                  " READ_DATA OUTPUT
*&-----
*&-----*
*&      Module DISPLAY_DATA OUTPUT
*&-----
*&-----*
*      text
*-----
*&-----*

MODULE display_data OUTPUT.
    CREATE OBJECT o_container
    EXPORTING
*      parent                =
*      container_name        = 'C1'
*      style                  =
*      lifetime               = lifetime_default

```

```

*      repid                        =
*      dynnr                        =
*      no_autodef_progid_dynnr     =
*  EXCEPTIONS
*      cntl_error                   = 1
*      cntl_system_error           = 2
*      create_error                 = 3
*      lifetime_error               = 4
*      lifetime_dynpro_dynpro_link = 5
*      others                       = 6
*
*
IF sy-subrc <> 0.
*  MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*           WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
ENDIF.

CREATE OBJECT o_grid
EXPORTING
*      i_shellstyle                 = 0
*      i_lifetime                   =
*      i_parent                     = o_container
*      i_appl_events                = space
*      i_parentdbg                  =
*      i_applogparent               =
*      i_graphicsparent             =
*      i_name                       =
*      i_fcat_complete              = space
*  EXCEPTIONS
*      error_cntl_create            = 1
*      error_cntl_init              = 2
*      error_cntl_link              = 3
*      error_dp_create              = 4
*      others                       = 5
*
*
IF sy-subrc <> 0.
*  MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*           WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
ENDIF.

*Register the Event Handler Methods

```

```
CREATE OBJECT o_event_rcvr.
```

```
*For Each Event
```

```
SET HANDLER : o_event_rcvr->handle_double_click FOR o_g  
rid,
```

```
o_event_rcvr->handle_toolbar FOR o_g  
rid,
```

```
o_event_rcvr->handle_user_command FOR o_g  
rid.
```

```
CALL METHOD o_grid->set_table_for_first_display
```

```
EXPORTING
```

```
* i_buffer_active =  
* i_bypassing_buffer =  
* i_consistency_check =  
i_structure_name = 'KNA1'  
* is_variant =  
* i_save =  
* i_default = 'X'  
* is_layout =  
* is_print =  
* it_special_groups =  
* it_toolbar_excluding =  
* it_hyperlink =  
* it_alv_graphics =  
* it_except_qinfo =  
* ir_salv_adapter =
```

```
CHANGING
```

```
it_outtab = it_kna1  
* it_fieldcatalog =  
* it_sort =  
* it_filter =
```

```
* EXCEPTIONS
```

```
* invalid_parameter_combination = 1  
* program_error = 2  
* too_many_lines = 3  
* others = 4
```

```
IF sy-subrc <> 0.
```

```
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
```

```
*              WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.  
ENDIF.
```

```
ENDMODULE.              " DISPLAY_DATA  OUTPUT
```

```
*&-----  
-----*  
*&      Module  STATUS_9000  OUTPUT  
*&-----  
-----*  
*      text  
*-----  
-----*
```

```
MODULE status_9000 OUTPUT.  
  SET PF-STATUS 'ZSTATUS'.
```

```
*  SET TITLEBAR 'xxx'.
```

```
ENDMODULE.              " STATUS_9000  OUTPUT
```

```
*&-----  
-----*  
*&      Module  USER_COMMAND_9000  INPUT  
*&-----  
-----*  
*      text  
*-----  
-----*
```

```
MODULE user_command_9000 INPUT.  
  CASE sy-ucomm.  
    WHEN 'BK'.  
      LEAVE TO SCREEN 0.  
  ENDCASE.
```

```
ENDMODULE.              " USER_COMMAND_9000  INPUT
```

8.00 ALV SECOND PROGRAM

```
*&-----  
-----*  
*& Report  ZRISHI_00_ALV_ALL
```

```

*&
*&-----
-----*
*&
*&
*&-----
-----*

```

REPORT zrishi_oo_alv_all.

```

DATA : o_container TYPE REF TO cl_gui_custom_container,
       o_grid      TYPE REF TO cl_gui_alv_grid,
       it_ekko     TYPE TABLE OF ekko.

```

**Define Event Handler Class*

CLASS lcl_event_receiver DEFINITION.

PUBLIC SECTION.

```

METHODS : handle_double_click FOR EVENT double_click
          OF cl_gui_alv_grid IMPORTING e_row e_column,

```

```

          handle_toolbar FOR EVENT toolbar
          OF cl_gui_alv_grid IMPORTING e_object,

```

```

          handle_user_command FOR EVENT user_command
          OF cl_gui_alv_grid IMPORTING e_ucomm.

```

ENDCLASS.

"LCL_EVENT_RECEIVER DEFINITION

**Implement Event Handler Class*

CLASS lcl_event_receiver IMPLEMENTATION.

METHOD handle_user_command.

```

DATA it_ekpo TYPE TABLE OF ekpo.

```

CASE e_ucomm.

WHEN 'GETAI'.

```

IF it_ekko IS NOT INITIAL .

```

```

SELECT * INTO TABLE it_ekpo FROM ekpo FOR ALL ENTRI
ES IN it_ekko

```

```

WHERE ebeln = it_ek

```

ko-ebeln.

```

CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'

```

EXPORTING

```
*          I_INTERFACE_CHECK                = ' '
*          I_BYPASSING_BUFFER               = ' '
*          I_BUFFER_ACTIVE                  = ' '
*          I_CALLBACK_PROGRAM                = ' '
*          I_CALLBACK_PF_STATUS_SET          = ' '
*          I_CALLBACK_USER_COMMAND           = ' '
*          I_CALLBACK_TOP_OF_PAGE            = ' '
*          I_CALLBACK_HTML_TOP_OF_PAGE       = ' '
*          I_CALLBACK_HTML_END_OF_LIST       = ' '
*          i_structure_name                  = 'EKPO'
*          I_BACKGROUND_ID                   = ' '
*          I_GRID_TITLE                      =
*          I_GRID_SETTINGS                   =
*          IS_LAYOUT                         =
*          IT_FIELDCAT                       =
*          IT_EXCLUDING                      =
*          IT_SPECIAL_GROUPS                 =
*          IT_SORT                           =
*          IT_FILTER                         =
*          IS_SEL_HIDE                       =
*          I_DEFAULT                         = 'X'
*          I_SAVE                            = '
,
*          IS_VARIANT                        =
*          IT_EVENTS                         = IT_
EVENTS
*          IT_EVENT_EXIT                     =
*          IS_PRINT                          =
*          IS_REPREP_ID                      =
*          I_SCREEN_START_COLUMN             = 0
*          I_SCREEN_START_LINE               = 0
*          I_SCREEN_END_COLUMN               = 0
*          I_SCREEN_END_LINE                 = 0
*          I_HTML_HEIGHT_TOP                 = 0
*          I_HTML_HEIGHT_END                 = 0
*          IT_ALV_GRAPHICS                   =
*          IT_HYPERLINK                      =
*          IT_ADD_FIELDCAT                   =
*          IT_EXCEPT_QINFO                  =
```



```

*          IR_SALV_FULLSCREEN_ADAPTER              =
*      IMPORTING
*          E_EXIT_CAUSED_BY_CALLER                  =
*          ES_EXIT_CAUSED_BY_USER                    =
TABLES
    t_outtab                                         = it_ekpo
*      EXCEPTIONS
*          PROGRAM_ERROR                            = 1
*          OTHERS                                    = 2
.
    IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*       WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
    ENDIF.
    ENDIF.
    ENDCASE.
ENDMETHOD.                                     "HANDLE_USER_COMMAND

METHOD handle_toolbar.
    DATA wa_button TYPE LINE OF ttb_button.
    wa_button-function = 'GETAI'.
    wa_button-text = 'GetAllItems'.
    APPEND wa_button TO e_object->mt_toolbar.

ENDMETHOD.                                     "HANDLE_TOOLBAR

METHOD handle_double_click.
    DATA : wa_ekko TYPE ekko,
            it_ekpo TYPE TABLE OF ekpo.
    READ TABLE it_ekko INTO wa_ekko INDEX e_row-index.

    IF e_column = 'EBELN'.
        SELECT * INTO TABLE it_ekpo FROM ekpo
            WHERE ebeln = wa_ekko-ebeln.

        CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
            EXPORTING
*          I_INTERFACE_CHECK                        = ' '
*          I_BYPASSING_BUFFER                      = ' '
*          I_BUFFER_ACTIVE                        = ' '

```

```

*          I_CALLBACK_PROGRAM                      = ' '
*          I_CALLBACK_PF_STATUS_SET                = ' '
*          I_CALLBACK_USER_COMMAND                 = ' '
*          I_CALLBACK_TOP_OF_PAGE                   = ' '
*          I_CALLBACK_HTML_TOP_OF_PAGE              = ' '
*          I_CALLBACK_HTML_END_OF_LIST              = ' '
i_structure_name                                = 'EKPO'
*          I_BACKGROUND_ID                         = ' '
*          I_GRID_TITLE                           =
*          I_GRID_SETTINGS                        =
*          IS_LAYOUT                              =
*          IT_FIELDCAT                             =
*          IT_EXCLUDING                           =
*          IT_SPECIAL_GROUPS                       =
*          IT_SORT                                 =
*          IT_FILTER                               =
*          IS_SEL_HIDE                             =
*          I_DEFAULT                               = 'X'
*          I_SAVE                                  = '
,
*          IS_VARIANT                             =
*          IT_EVENTS                              = IT_
EVENTS
*          IT_EVENT_EXIT                          =
*          IS_PRINT                                =
*          IS_REPREP_ID                            =
*          I_SCREEN_START_COLUMN                   = 0
*          I_SCREEN_START_LINE                     = 0
*          I_SCREEN_END_COLUMN                     = 0
*          I_SCREEN_END_LINE                       = 0
*          I_HTML_HEIGHT_TOP                       = 0
*          I_HTML_HEIGHT_END                       = 0
*          IT_ALV_GRAPHICS                         =
*          IT_HYPERLINK                           =
*          IT_ADD_FIELDCAT                         =
*          IT_EXCEPT_QINFO                       =
*          IR_SALV_FULLSCREEN_ADAPTER              =
*          IMPORTING
*          E_EXIT_CAUSED_BY_CALLER                 =
*          ES_EXIT_CAUSED_BY_USER                  =

```

```

TABLES
    t_outtab                                = it_ekpo
*
*      EXCEPTIONS
*      PROGRAM_ERROR                        = 1
*      OTHERS                              = 2
*
*
*      IF sy-subrc <> 0.
*      MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*      WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
*      ENDIF.
*      ENDIF.

*      ENDMETHOD.                                "HANDLE_DOUBLE_CLICK
*      ENDCLASS.                                "LCL_EVENT_RECEIVER IMPLEMENTATI
ON

DATA o_event_rcvr TYPE REF TO lcl_event_receiver.

START-OF-SELECTION.
    CALL SCREEN 9000.

*&-----
*&-----*
*&      Module  DISPLAY_ALV_9000  OUTPUT
*&-----
*&-----*
MODULE display_alv_9000 OUTPUT.

    CREATE OBJECT o_container
    EXPORTING
*      parent                                =
*      container_name                        = 'C1'
*      style                                =
*      lifetime                              = lifetime_default
*      repid                                =
*      dynnr                                =
*      no_autodef_progid_dynnr              =
*      EXCEPTIONS
*      cntl_error                            = 1

```

```

*      cntl_system_error          = 2
*      create_error                = 3
*      lifetime_error              = 4
*      lifetime_dynpro_dynpro_link = 5
*      others                      = 6

```

```

IF sy-subrc <> 0.

```

```

* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*           WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.

```

```

ENDIF.

```

```

CREATE OBJECT o_grid

```

```

EXPORTING

```

```

*      i_shellstyle          = 0
*      i_lifetime            =
*      i_parent              = o_container
*      i_appl_events         = space
*      i_parentdbg           =
*      i_applogparent        =
*      i_graphicsparent      =
*      i_name                =
*      i_fcat_complete       = space
* EXCEPTIONS
*      error_cntl_create     = 1
*      error_cntl_init       = 2
*      error_cntl_link       = 3
*      error_dp_create       = 4
*      others                = 5

```

```

IF sy-subrc <> 0.

```

```

* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*           WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.

```

```

ENDIF.

```

```

*Register the Event Handler Methods

```

```

CREATE OBJECT o_event_rcvr.

```

```

*For Each Event

```

```

SET HANDLER : o_event_rcvr->handle_double_click      FOR o_g

```

```

rid,
            o_event_rcvr->handle_toolbar          FOR o_g
rid,
            o_event_rcvr->handle_user_command     FOR o_g
rid.

```

```

CALL METHOD o_grid->set_table_for_first_display
EXPORTING

```

```

*      i_buffer_active                =
*      i_bypassing_buffer             =
*      i_consistency_check            =
*      i_structure_name               = 'EKKO'
*      is_variant                     =
*      i_save                          =
*      i_default                      = 'X'
*      is_layout                      =
*      is_print                       =
*      it_special_groups              =
*      it_toolbar_excluding           =
*      it_hyperlink                   =
*      it_alv_graphics                =
*      it_except_qinfo                =
*      ir_salv_adapter                =

```

```

CHANGING

```

```

*      it_outtab                      = it_ekko
*      it_fieldcatalog                =
*      it_sort                        =
*      it_filter                      =

```

```

*      EXCEPTIONS
*      invalid_parameter_combination = 1
*      program_error                 = 2
*      too_many_lines                 = 3
*      others                         = 4

```

```

IF sy-subrc <> 0.

```

```

*      MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*              WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.

```

```

ENDIF.

```

```
ENDMODULE. " DISPLAY_ALV_9000 OUTPUT
```

```
*&-----*  
-----*
```

```
*& Module READ_DATA_9000 OUTPUT
```

```
*&-----*  
-----*
```

```
MODULE read_data_9000 OUTPUT.
```

```
  SELECT * INTO TABLE it_ekko FROM ekko UP TO 20 ROWS  
        WHERE bstyp = 'F'.
```

```
ENDMODULE. " READ_DATA_9000 OUTPUT
```

```
*&-----*  
-----*
```

```
*& Module STATUS_9000 OUTPUT
```

```
*&-----*  
-----*
```

```
MODULE status_9000 OUTPUT.
```

```
  SET PF-STATUS 'ZSTATUS'.
```

```
  * SET TITLEBAR 'xxx'.
```

```
ENDMODULE. " STATUS_9000 OUTPUT
```

```
*&-----*  
-----*
```

```
*& Module USER_COMMAND_9000 INPUT
```

```
*&-----*  
-----*
```

```
MODULE user_command_9000 INPUT.
```

```
  CASE sy-ucomm.
```

```
    WHEN 'BK'.
```

```
      LEAVE TO SCREEN 0.
```

```
  ENDCASE.
```

```
ENDMODULE. " USER_COMMAND_9000 INPUT
```

```
9.00 ALV EVENT HANDLING
```

```
*&-----*  
-----*
```

```
*& Report ZRISHI_00_ALV_EVENT_HANDLING
```

```

*&
*&-----
-----*
*&
*&
*&-----
-----*

```

```

REPORT  zrishi_oo_alv_event_handling.
DATA : o_container TYPE REF TO cl_gui_custom_container,
       o_grid      TYPE REF TO cl_gui_alv_grid,
       it_kna1     TYPE TABLE OF kna1.

*-----
-----*
*          CLASS lcl_event_receiver DEFINITION
*-----
-----*
*
*-----
-----*

```

```

CLASS lcl_event_receiver DEFINITION.
  PUBLIC SECTION.
    METHODS : handle_double_click FOR EVENT double_click
              OF cl_gui_alv_grid IMPORTING e_row e_column,

              handle_toolbar FOR EVENT toolbar
              OF cl_gui_alv_grid IMPORTING e_object,

              handle_user_command FOR EVENT user_command
              OF cl_gui_alv_grid IMPORTING e_ucomm.

ENDCLASS.                                "lcl_event_receiver DEFINITION

```

**Implement Event Handler Class*

```

CLASS lcl_event_receiver IMPLEMENTATION.

  METHOD handle_user_command.
    DATA it_lfa1 TYPE TABLE OF lfa1.
    CASE e_ucomm.
      WHEN 'GETAI'.

```

```

        IF it_kna1 IS NOT INITIAL .
            SELECT * INTO TABLE it_lfa1 FROM lfa1 FOR ALL ENTRI
ES IN it_kna1
                                WHERE kunnr = it_kn
a1-kunnr.

```

```

CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
EXPORTING

```

```

*          I_INTERFACE_CHECK                = ' '
*          I_BYPASSING_BUFFER               = ' '
*          I_BUFFER_ACTIVE                  = ' '
*          I_CALLBACK_PROGRAM                = ' '
*          I_CALLBACK_PF_STATUS_SET          = ' '
*          I_CALLBACK_USER_COMMAND           = ' '
*          I_CALLBACK_TOP_OF_PAGE            = ' '
*          I_CALLBACK_HTML_TOP_OF_PAGE       = ' '
*          I_CALLBACK_HTML_END_OF_LIST       = ' '
*          i_structure_name                  = 'LFA1'
*          I_BACKGROUND_ID                   = ' '
*          I_GRID_TITLE                      =
*          I_GRID_SETTINGS                   =
*          IS_LAYOUT                          =
*          IT_FIELDCAT                       =
*          IT_EXCLUDING                      =
*          IT_SPECIAL_GROUPS                 =
*          IT_SORT                           =
*          IT_FILTER                         =
*          IS_SEL_HIDE                       =
*          I_DEFAULT                         = 'X'
*          I_SAVE                            = '
,
*          IS_VARIANT                        =
*          IT_EVENTS                         = IT_
EVENTS
*          IT_EVENT_EXIT                    =
*          IS_PRINT                          =
*          IS_REPREP_ID                     =
*          I_SCREEN_START_COLUMN             = 0
*          I_SCREEN_START_LINE               = 0
*          I_SCREEN_END_COLUMN              = 0
*          I_SCREEN_END_LINE                 = 0

```



```

*          I_HTML_HEIGHT_TOP                        = 0
*          I_HTML_HEIGHT_END                        = 0
*          IT_ALV_GRAPHICS                          =
*          IT_HYPERLINK                            =
*          IT_ADD_FIELDCAT                          =
*          IT_EXCEPT_QINFO                        =
*          IR_SALV_FULLSCREEN_ADAPTER               =
*          IMPORTING
*          E_EXIT_CAUSED_BY_CALLER                  =
*          ES_EXIT_CAUSED_BY_USER                   =

TABLES
    t_outtab                                = it_lfa1
*          EXCEPTIONS
*          PROGRAM_ERROR                        = 1
*          OTHERS                              = 2
.
    IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*       WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
    ENDIF.
    ENDIF.
    ENDCASE.
ENDMETHOD.                                "handle_user_command

METHOD handle_toolbar.
    DATA wa_button TYPE LINE OF ttb_button.
    wa_button-function = 'GETAI'.
    wa_button-text = 'GetAllItems'.
    APPEND wa_button TO e_object->mt_toolbar.

ENDMETHOD.                                "handle_toolbar

METHOD handle_double_click.
    DATA : wa_kna1 TYPE kna1,
            it_lfa1 TYPE TABLE OF lfa1.
    READ TABLE it_kna1 INTO wa_kna1 INDEX e_row-index.

    IF e_column = 'KUNNR'.
        SELECT * INTO TABLE it_lfa1 FROM lfa1
            WHERE kunnr = wa_kna1-kunnr.

```

CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
EXPORTING

```

*          I_INTERFACE_CHECK                = ' '
*          I_BYPASSING_BUFFER               = ' '
*          I_BUFFER_ACTIVE                  = ' '
*          I_CALLBACK_PROGRAM               = ' '
*          I_CALLBACK_PF_STATUS_SET         = ' '
*          I_CALLBACK_USER_COMMAND         = ' '
*          I_CALLBACK_TOP_OF_PAGE          = ' '
*          I_CALLBACK_HTML_TOP_OF_PAGE     = ' '
*          I_CALLBACK_HTML_END_OF_LIST     = ' '
*          i_structure_name                 = 'LFA1'
*          I_BACKGROUND_ID                 = ' '
*          I_GRID_TITLE                    =
*          I_GRID_SETTINGS                 =
*          IS_LAYOUT                       =
*          IT_FIELDCAT                     =
*          IT_EXCLUDING                    =
*          IT_SPECIAL_GROUPS               =
*          IT_SORT                         =
*          IT_FILTER                       =
*          IS_SEL_HIDE                     =
*          I_DEFAULT                       = 'X'
*          I_SAVE                          = '
,
*          IS_VARIANT                      =
*          IT_EVENTS                       = IT_
EVENTS
*          IT_EVENT_EXIT                   =
*          IS_PRINT                        =
*          IS_REPREP_ID                   =
*          I_SCREEN_START_COLUMN           = 0
*          I_SCREEN_START_LINE             = 0
*          I_SCREEN_END_COLUMN             = 0
*          I_SCREEN_END_LINE               = 0
*          I_HTML_HEIGHT_TOP               = 0
*          I_HTML_HEIGHT_END               = 0
*          IT_ALV_GRAPHICS                 =
*          IT_HYPERLINK                    =

```

```

*          IT_ADD_FIELDCAT                      =
*          IT_EXCEPT_QINFO                    =
*          IR_SALV_FULLSCREEN_ADAPTER           =
*          IMPORTING
*          E_EXIT_CAUSED_BY_CALLER              =
*          ES_EXIT_CAUSED_BY_USER               =

TABLES
    t_outtab                                = it_lfal

*          EXCEPTIONS
*          PROGRAM_ERROR                      = 1
*          OTHERS                            = 2
.

IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
ENDIF.
ENDIF.

ENDMETHOD.                                "handle_double_click
ENDCLASS.                                "lcl_event_receiver IMPLEMENTATI
ON

DATA o_event_rcvr TYPE REF TO lcl_event_receiver.

START-OF-SELECTION.
    CALL SCREEN 9000.

*&-----
*&-----*
*&      Module  DISPLAY_ALV_9000  OUTPUT
*&-----
*&-----*

MODULE display_alv_9000 OUTPUT.

    CREATE OBJECT o_container
    EXPORTING
*      parent                      =
*      container_name              = 'C1'
*      style                      =

```

```

*      lifetime                = lifetime_default
*      repid                   =
*      dynnr                   =
*      no_autodef_progid_dynnr =
*  EXCEPTIONS
*      cntl_error               = 1
*      cntl_system_error       = 2
*      create_error             = 3
*      lifetime_error           = 4
*      lifetime_dynpro_dynpro_link = 5
*      others                   = 6

```

IF sy-subrc <> 0.

```

* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*           WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.

```

ENDIF.

CREATE OBJECT o_grid

EXPORTING

```

*      i_shellstyle            = 0
*      i_lifetime              =
*      i_parent                = o_container
*      i_appl_events           = space
*      i_parentdbg             =
*      i_applogparent          =
*      i_graphicsparent        =
*      i_name                   =
*      i_fcat_complete         = space
*  EXCEPTIONS
*      error_cntl_create       = 1
*      error_cntl_init         = 2
*      error_cntl_link         = 3
*      error_dp_create         = 4
*      others                  = 5

```

IF sy-subrc <> 0.

```

* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*           WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.

```

ENDIF.

**Register the Event Handler Methods*

CREATE OBJECT o_event_rcvr.

**For Each Event*

SET HANDLER : o_event_rcvr->handle_double_click **FOR** o_g
rid,

o_event_rcvr->handle_toolbar **FOR** o_g
rid,

o_event_rcvr->handle_user_command **FOR** o_g
rid.

CALL METHOD o_grid->set_table_for_first_display

EXPORTING

* i_buffer_active =
* i_bypassing_buffer =
* i_consistency_check =
i_structure_name = 'KNA1'
* is_variant =
* i_save =
* i_default = 'X'
* is_layout =
* is_print =
* it_special_groups =
* it_toolbar_excluding =
* it_hyperlink =
* it_alv_graphics =
* it_except_qinfo =
* ir_salv_adapter =

CHANGING

it_outtab = it_kna1
* it_fieldcatalog =
* it_sort =
* it_filter =

* **EXCEPTIONS**

* invalid_parameter_combination = 1
* program_error = 2
* too_many_lines = 3
* others = 4

```

IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
ENDIF.

ENDMODULE. " DISPLAY_ALV_9000 OUTPUT

*&-----*
*& Module READ_DATA_9000 OUTPUT
*&-----*
MODULE read_data_9000 OUTPUT.
  SELECT * INTO TABLE it_kna1 FROM kna1 UP TO 20 ROWS
  WHERE land1 = 'IN'.
ENDMODULE. " READ_DATA_9000 OUTPUT

*&-----*
*& Module STATUS_9000 OUTPUT
*&-----*
MODULE status_9000 OUTPUT.
  SET PF-STATUS 'ZSTATUS'.
  * SET TITLEBAR 'xxx'.
ENDMODULE. " STATUS_9000 OUTPUT

*&-----*
*& Module USER_COMMAND_9000 INPUT
*&-----*
MODULE user_command_9000 INPUT.
  CASE sy-ucomm.
    WHEN 'BK'.
      LEAVE TO SCREEN 0.
  ENDCASE.
ENDMODULE. " USER_COMMAND_9000 INPUT

```

10. 00 ALV EVENT HANDLING ALL

REPORT zrishi_oo_alv_events.

DATA : o_container TYPE REF TO cl_gui_custom_container,
o_grid TYPE REF TO cl_gui_alv_grid,
it_ekko TYPE TABLE OF ekko,
o_splitter TYPE REF TO cl_gui_splitter_container,
o_main TYPE REF TO cl_gui_container,
o_header TYPE REF TO cl_gui_container,
o_document TYPE REF TO cl_dd_document.

**Define Event Handler Class*

CLASS lcl_event_receiver DEFINITION.

PUBLIC SECTION.

METHODS : handle_double_click FOR EVENT double_click
OF cl_gui_alv_grid IMPORTING e_row e_column,

handle_toolbar FOR EVENT toolbar
OF cl_gui_alv_grid IMPORTING e_object,

handle_user_command FOR EVENT user_command
OF cl_gui_alv_grid IMPORTING e_ucomm,

handle_top_of_page FOR EVENT top_of_page
OF cl_gui_alv_grid,
handle_before_user_command FOR EVENT
before_user_command OF cl_gui_alv_grid
IMPORTING e_ucomm .

ENDCLASS.

"lcl_event_receiver DEFINITION

**Implement Event Handler Class*

CLASS lcl_event_receiver IMPLEMENTATION.

METHOD handle_before_user_command.

DATA : it_index_rows TYPE lvc_t_row,
wa TYPE LINE OF lvc_t_row,
it_selected_ekko TYPE TABLE OF ekko,
wa_ekko TYPE ekko.

```

CASE e_ucomm.
  WHEN '&PC'.
    MESSAGE 'I am Executed before U' TYPE 'I'.

*Returns Only the Indexs of the Selected Rows
  CALL METHOD o_grid->get_selected_rows
  IMPORTING
    et_index_rows = it_index_rows
*
  et_row_no      =
  .

*Get the Content by reading for Each Index In it_index_rows

  LOOP AT it_index_rows INTO wa.
    READ TABLE it_ekko INTO
      wa_ekko INDEX wa-index.
    APPEND wa_ekko TO it_selected_ekko.
  ENDLOOP.

  CALL METHOD cl_gui_frontend_services=>gui_download
  EXPORTING
*    bin_filesize          =
*      filename            = 'F:7AM_SELECTED.TXT'
*    filetype              = 'ASC'
*    append                 = SPACE
*    write_field_separator = SPACE
*    header                 = '00'
*    trunc_trailing_blanks = SPACE
*    write_lf               = 'X'
*    col_select             = SPACE
*    col_select_mask       = SPACE
*    dat_mode              = SPACE
*    confirm_overwrite     = SPACE
*    no_auth_check         = SPACE
*    codepage              = SPACE
*    ignore_cerr           = ABAP_TRUE
*    replacement           = '#'
*    write_bom             = SPACE
*    trunc_trailing_blanks_eol = 'X'
*    wkl_n_format          = SPACE

```



```

*      wk1_n_size          = SPACE
*      wk1_t_format        = SPACE
*      wk1_t_size          = SPACE
*      show_transfer_status = 'X'
*      fieldnames           =
*      write_lf_after_last_line = 'X'
*      IMPORTING
*      filelength           =
*
*      CHANGING
*      data_tab              = it_selected_ekko
*      EXCEPTIONS
*      file_write_error      = 1
*      no_batch              = 2
*      gui_refuse_filetransfer = 3
*      invalid_type          = 4
*      no_authority          = 5
*      unknown_error         = 6
*      header_not_allowed    = 7
*      separator_not_allowed = 8
*      filesize_not_allowed  = 9
*      header_too_long       = 10
*      dp_error_create       = 11
*      dp_error_send         = 12
*      dp_error_write        = 13
*      unknown_dp_error      = 14
*      access_denied         = 15
*      dp_out_of_memory      = 16
*      disk_full             = 17
*      dp_timeout            = 18
*      file_not_found        = 19
*      dataprovider_exception = 20
*      control_flush_error   = 21
*      not_supported_by_gui  = 22
*      error_no_gui          = 23
*      others                 = 24
*
*      .
*      IF sy-subrc = 0.
*          MESSAGE 'selected Lines are Successfully Downloaded'
TYPE 'S'.
*      ENDIF.

```

```

        ULINE.

        CALL METHOD o_grid->set_user_command
        EXPORTING
            i_ucomm = space.  "Std.Function Decactivated

    ENDCASE.
ENDMETHOD .  "handle_before_user_command

METHOD handle_top_of_page.
    CALL METHOD o_document->add_picture
    EXPORTING
        picture_id      = 'ENJOYSAP_LOGO'
*       width           =
*       alternative_text =
*       tabindex        =
    .

    CALL METHOD o_document->add_text
    EXPORTING
        text            = 'List Of Purchasing Documents'
*       text_table      =
*       fix_lines       =
*       sap_style        =
*       sap_color        =
*       sap_fontsize     =
*       sap_fontstyle    =
*       sap_emphasis     =
*       style_class      =
*   CHANGING
*       document         =
    .

    CALL METHOD o_document->add_picture
    EXPORTING
        picture_id      = 'ENJOYSAP_LOGO'
*       width           =
*       alternative_text =
*       tabindex        =
    .

```

```

CALL METHOD o_document->display_document
EXPORTING
*   reuse_control      =
*   reuse_registration =
*   container          =
      parent          = o_header
* EXCEPTIONS
*   html_display_error = 1
*   others              = 2
      .

IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*           WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
ENDIF.

ENDMETHOD.                                "handle_top_of_page

METHOD handle_user_command.
DATA it_ekpo TYPE TABLE OF   ekpo.
CASE e_ucomm.
  WHEN 'GETAI'.
    IF it_ekko IS NOT INITIAL .
      SELECT * INTO TABLE it_ekpo FROM ekpo FOR ALL
ENTRIES IN it_ekko
                                                    WHERE ebeln =
it_ekko-ebeln.
      CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
EXPORTING
*           I_INTERFACE_CHECK           = ' '
*           I_BYPASSING_BUFFER          = ' '
*           I_BUFFER_ACTIVE             = ' '
*           I_CALLBACK_PROGRAM          = ' '
*           I_CALLBACK_PF_STATUS_SET    = ' '
*           I_CALLBACK_USER_COMMAND     = ' '
*           I_CALLBACK_TOP_OF_PAGE      = ' '
*           I_CALLBACK_HTML_TOP_OF_PAGE = ' '
*           I_CALLBACK_HTML_END_OF_LIST = ' '
      i_structure_name = 'EKPO'
*           I_BACKGROUND_ID            = ' '

```

```

*          I_GRID_TITLE                      =
*          I_GRID_SETTINGS                  =
*          IS_LAYOUT                        = IS_LAYOUT
*          IT_FIELDCAT                      =
*          IT_EXCLUDING                     =
*          IT_SPECIAL_GROUPS                =
*          IT_SORT                          =
*          IT_FILTER                        =
*          IS_SEL_HIDE                      =
*          I_DEFAULT                        = 'X'
*          I_SAVE                           = ' '
*          IS_VARIANT                       =
*          IT_EVENTS                        = IT_EVENTS
*          IT_EVENT_EXIT                    =
*          IS_PRINT                         =
*          IS_REPREP_ID                     =
*          I_SCREEN_START_COLUMN            = 0
*          I_SCREEN_START_LINE              = 0
*          I_SCREEN_END_COLUMN              = 0
*          I_SCREEN_END_LINE                = 0
*          I_HTML_HEIGHT_TOP                = 0
*          I_HTML_HEIGHT_END                = 0
*          IT_ALV_GRAPHICS                  =
*          IT_HYPERLINK                     =
*          IT_ADD_FIELDCAT                  =
*          IT_EXCEPT_QINFO                 =
*          IR_SALV_FULLSCREEN_ADAPTER        =
*      IMPORTING
*          E_EXIT_CAUSED_BY_CALLER          =
*          ES_EXIT_CAUSED_BY_USER           =

```

TABLES

```

*          t_outtab                        = it_ekpo
*      EXCEPTIONS
*          PROGRAM_ERROR                    = 1
*          OTHERS                           = 2

```

IF sy-subrc <> 0.

```

*  MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*  WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
ENDIF.

```

```

        ENDIF.
    ENDCASE.
ENDMETHOD.                                "handle_user_command

METHOD handle_toolbar.
    DATA wa_button TYPE LINE OF ttb_button.
    wa_button-function = 'GETAI'.
    wa_button-text = 'GetAllItems'.
    APPEND wa_button TO e_object->mt_toolbar.

ENDMETHOD.                                "handle_toolbar

METHOD handle_double_click.
    DATA : wa_ekko TYPE ekko,
            it_ekpo TYPE TABLE OF ekpo.
    READ TABLE it_ekko INTO wa_ekko INDEX e_row-index.

    IF e_column = 'EBELN'.
        SELECT * INTO TABLE it_ekpo FROM ekpo
            WHERE ebeln = wa_ekko-ebeln.

        CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
            EXPORTING
*               I_INTERFACE_CHECK                = ' '
*               I_BYPASSING_BUFFER              = ' '
*               I_BUFFER_ACTIVE                 = ' '
*               I_CALLBACK_PROGRAM              = ' '
*               I_CALLBACK_PF_STATUS_SET        = ' '
*               I_CALLBACK_USER_COMMAND         = ' '
*               I_CALLBACK_TOP_OF_PAGE          = ' '
*               I_CALLBACK_HTML_TOP_OF_PAGE     = ' '
*               I_CALLBACK_HTML_END_OF_LIST     = ' '
                i_structure_name                 = 'EKPO'
*               I_BACKGROUND_ID                 = ' '
*               I_GRID_TITLE                    =
*               I_GRID_SETTINGS                 =
*               IS_LAYOUT                       =
*               IT_FIELDCAT                     =
*               IT_EXCLUDING                    =
*               IT_SPECIAL_GROUPS               =

```

```

*          IT_SORT                      =
*          IT_FILTER                    =
*          IS_SEL_HIDE                  =
*          I_DEFAULT                    = 'X'
*          I_SAVE                       = ' '
*          IS_VARIANT                   =
*          IT_EVENTS                    = IT_EVENTS
*          IT_EVENT_EXIT                =
*          IS_PRINT                     =
*          IS_REPREP_ID                 =
*          I_SCREEN_START_COLUMN        = 0
*          I_SCREEN_START_LINE          = 0
*          I_SCREEN_END_COLUMN          = 0
*          I_SCREEN_END_LINE            = 0
*          I_HTML_HEIGHT_TOP            = 0
*          I_HTML_HEIGHT_END            = 0
*          IT_ALV_GRAPHICS              =
*          IT_HYPERLINK                 =
*          IT_ADD_FIELDCAT              =
*          IT_EXCEPT_QINFO            =
*          IR_SALV_FULLSCREEN_ADAPTER    =
*      IMPORTING
*          E_EXIT_CAUSED_BY_CALLER      =
*          ES_EXIT_CAUSED_BY_USER       =

```

TABLES

```

      t_outtab                          = it_ekpo
*      EXCEPTIONS
*          PROGRAM_ERROR                 = 1
*          OTHERS                       = 2

```

IF sy-subrc <> 0.

```

* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.

```

ENDIF.

ENDIF.

ENDMETHOD.

ENDCLASS.

IMPLEMENTATION

```

      "handle_double_click
      "lcl_event_receiver

```

```
DATA o_event_rcvr TYPE REF TO lcl_event_receiver.
```

```
START-OF-SELECTION.
```

```
  CALL SCREEN 9000.
```

```
*&-----*  
-----*  
*&      Module  DISPLAY_ALV_9000  OUTPUT  
*&-----*  
-----*
```

```
MODULE display_alv_9000 OUTPUT.
```

```
  DATA is_layout TYPE lvc_s_layo.
```

```
  is_layout-sel_mode = 'D'. "Allows Multi Line Selection
```

```
  CREATE OBJECT o_grid
```

```
    EXPORTING
```

```
*    i_shellstyle      = 0  
*    i_lifetime        =  
*    i_parent          = o_main  
*    i_appl_events     = space  
*    i_parentdbg       =  
*    i_applogparent    =  
*    i_graphicsparent  =  
*    i_name            =  
*    i_fcat_complete   = space  
* EXCEPTIONS  
*    error_cntl_create = 1  
*    error_cntl_init   = 2  
*    error_cntl_link   = 3  
*    error_dp_create   = 4  
*    others            = 5
```

```
  IF sy-subrc <> 0.
```

```
*    MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO  
*              WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
```

```
  ENDIF.
```

**Register the Event Handler Methods*

```
CREATE OBJECT o_event_rcvr.
```

**For Each Event*

```
SET HANDLER : o_event_rcvr->handle_double_click FOR
o_grid,
                o_event_rcvr->handle_toolbar FOR
o_grid,
                o_event_rcvr->handle_user_command FOR
o_grid,
                o_event_rcvr->handle_top_of_page FOR
o_grid,
                o_event_rcvr->handle_before_user_command FOR
o_grid.
```

```
CREATE OBJECT o_document
```

.

**To Trigger the List Processing Event -> 'TOP_OF_PAGE'*

```
CALL METHOD o_grid->list_processing_events
```

```
EXPORTING
```

```
    i_event_name      = 'TOP_OF_PAGE'
```

```
    i_dyndoc_id       = o_document
```

```
*    is_subtottxt_info =
```

```
*    ip_subtot_line   =
```

```
*    i_table_index    =
```

```
*    CHANGING
```

```
*    c_subtottxt      =
```

.

```
DATA : it_fieldcat TYPE lvc_t_fcat,
      wa_fieldcat TYPE LINE OF lvc_t_fcat.
```

```
CALL FUNCTION 'LVC_FIELDCATALOG_MERGE'
```

```
EXPORTING
```

```
*    I_BUFFER_ACTIVE      =
```

```
    i_structure_name      = 'EKKO'
```

```
*    I_CLIENT_NEVER_DISPLAY = 'X'
```



```

*   I_BYPASSING_BUFFER           =
*   I_INTERNAL_TABNAME          =
    CHANGING
        ct_fieldcat              = it_fieldcat
* EXCEPTIONS
*   INCONSISTENT_INTERFACE      = 1
*   PROGRAM_ERROR               = 2
*   OTHERS                      = 3
.
IF sy-subrc <> 0.
* Implement suitable error handling here
ENDIF.

CLEAR wa_fieldcat.
wa_fieldcat-emphasize = 'C71'.
MODIFY it_fieldcat FROM wa_fieldcat TRANSPORTING emphasize
                        WHERE fieldname = 'LIFNR'.
CALL METHOD o_grid->set_table_for_first_display
    EXPORTING
*   i_buffer_active              =
*   i_bypassing_buffer           =
*   i_consistency_check          =
*   i_structure_name             = 'EKKO'
*   is_variant                   =
*   i_save                       =
*   i_default                    = 'X'
        is_layout                = is_layout
*   is_print                     =
*   it_special_groups            =
*   it_toolbar_excluding         =
*   it_hyperlink                 =
*   it_alv_graphics              =
*   it_except_qinfo              =
*   ir_salv_adapter              =
    CHANGING
        it_outtab                = it_ekko
        it_fieldcatalog           = it_fieldcat
*   it_sort                      =
*   it_filter                    =
* EXCEPTIONS

```

```

*      invalid_parameter_combination = 1
*      program_error                  = 2
*      too_many_lines                 = 3
*      others                         = 4
.
IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*           WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
ENDIF.

```

```

ENDMODULE.                                " DISPLAY_ALV_9000 OUTPUT

```

```

*&-----
*-----*
*&      Module READ_DATA_9000 OUTPUT
*&-----
*-----*

```

```

MODULE read_data_9000 OUTPUT.
  SELECT * INTO TABLE it_ekko FROM ekko UP TO 20 ROWS
        WHERE bstyp = 'F'.
ENDMODULE.                                " READ_DATA_9000 OUTPUT

```

```

*&-----
*-----*
*&      Module STATUS_9000 OUTPUT
*&-----
*-----*

```

```

MODULE status_9000 OUTPUT.
  SET PF-STATUS 'ZSTATUS'.
*   SET TITLEBAR 'xxx'.

```

```

ENDMODULE.                                " STATUS_9000 OUTPUT

```

```

*&-----
*-----*
*&      Module USER_COMMAND_9000 INPUT
*&-----
*-----*

```

```

MODULE user_command_9000 INPUT.

```

```

CASE sy-ucomm.
    WHEN 'BK'.
        LEAVE TO SCREEN 0.
ENDCASE.
ENDMODULE.                                " USER_COMMAND_9000 INPUT

*&-----
-----*
*&      Module  SPLIT_CONTAINER_9000  OUTPUT
*&-----
-----*
MODULE split_container_9000 OUTPUT.
    CREATE OBJECT o_container
    EXPORTING
        *      parent                =
            container_name            = 'C1'
        *      style                  =
        *      lifetime               = lifetime_default
        *      repid                  =
        *      dynnr                  =
        *      no_autodef_progid_dynnr =
        *      EXCEPTIONS
        *      cntl_error              = 1
        *      cntl_system_error       = 2
        *      create_error            = 3
        *      lifetime_error          = 4
        *      lifetime_dynpro_dynpro_link = 5
        *      others                  = 6
        .
    IF sy-subrc <> 0.
        * MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
        *           WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
    ENDIF.

    CREATE OBJECT o_splitter
    EXPORTING
        *      link_dynnr            =
        *      link_repid             =
        *      shellstyle             =
        *      left                   =

```

```

*      top                =
*      width              =
*      height             =
*      metric              = cntl_metric_dynpro
*      align              = 15
*      parent              = o_container
*      rows                = 2
*      columns             = 1
*      no_autodef_progid_dynnr =
*      name                =
*      EXCEPTIONS
*      cntl_error          = 1
*      cntl_system_error = 2
*      others              = 3
*
*      .
IF sy-subrc <> 0.
*      MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*              WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
ENDIF.

CALL METHOD o_splitter->get_container
EXPORTING
    row          = 1
    column       = 1
RECEIVING
    container    = o_header.

CALL METHOD o_splitter->get_container
EXPORTING
    row          = 2
    column       = 1
RECEIVING
    container    = o_main.

CALL METHOD o_splitter->set_row_height
EXPORTING
    id           = 1
"Row1
    height       = 25
*      IMPORTING

```

